

Uniwersytet Warszawski
Wydział Matematyki, Informatyki i Mechaniki

Maciej Wielgus

Nr albumu: 262778

Perona-Malik equation and its numerical properties

Praca licencjacka
na kierunku MATEMATYKA

Praca wykonana pod kierunkiem
dra Pawła Bechlera
Instytut Matematyki Stosowanej i Mechaniki

Lipiec 2010

Abstract

This work concerns the Perona-Malik equation, which plays essential role in image processing. The first part gives a survey of results on existence, uniqueness and stability of solutions, the second part introduces discretisations of equation and deals with an analysis of discrete problem. In the last part I present some numerical results, in particular with algorithms applied to real images.

Table of contents

Introduction	5
1. The Perona-Malik equation	7
1.1. Gaussian diffusion in image processing	7
1.2. Introducing the Perona-Malik equation	8
1.3. Investigation of well-posedness	9
1.4. Regularizations	11
2. Discretizations of the problem	13
2.1. Semidiscrete and discrete models	13
2.2. Well-posedness of discrete models	15
3. Numerical results	17
3.1. Comparison between Perona-Malik and Gaussian filters	17
3.2. Different diffusivity functions	18
3.3. 2D data filtration and evolution	19
A. List of implemented diffusion algorithms	23
Bibliography	25

Introduction

The Perona-Malik model, first proposed in 1987 [PM], is a nonlinear partial diffusion equation that uses an inhomogeneous diffusivity coefficient. It is widely used in image processing for purposes like smoothing, restoration, segmentation, filtering or detecting edges.

It was shown by Kawohl and Kutev that the equation may have no global weak solutions in C^1 [KK]. There are several methods which deal with this problem by regularizing the equation, both in space and time (see for example [CLMC]).

Although the basic model is ill-posed, its discretizations are found to be stable. This fact is sometimes referred to as the Perona-Malik paradox [Kit]. The explanation for these observations was given by Weickert and Benhamouda [WB], who investigated the regularizing effect of a standard finite difference discretization.

The following work presents an analysis of the Perona-Malik equation, surveys the results on ill-posedness of the continuous problem and its discrete versions. I also present numerical results, compare different versions of the algorithm and demonstrate the evolution of a picture under the Perona-Malik filter. As I present the Perona-Malik model from the perspective of image processing, I usually bound myself to the two-dimensional version of the equation, or, for clarity of demonstration, show the results in one-dimension. I do not include proofs as they are to be found in the publications listed in the bibliography.

All implementations are done in the Octave/Matlab environment.

Chapter 1

The Perona-Malik equation

1.1. Gaussian diffusion in image processing

The diffusion equation is a partial differential equation often arising from physical phenomena. It is usually presented in a general form

$$\partial_t u(x, t) = \nabla(C(u, x, t) \nabla u(x, t)) \quad (1.1)$$

where the coefficient C is called *diffusivity* and, depending on the model, can be a scalar, scalar function of coordinates (inhomogeneity) or a tensor (anisotropy). Moreover, the equation becomes nonlinear if the diffusivity coefficient depends on the solution u .

For a constant $C \equiv 1$ we obtain a simple linear model of the heat equation, which leads to the following Cauchy problem

$$\begin{cases} u_t = \Delta u & \text{for } x \in \mathbb{R}^n, t > 0 \\ u(x, 0) = f(x) & \text{for } x \in \mathbb{R}^n \end{cases} \quad (1.2)$$

Theorem 1.1 ([Strzel]) *If $f \in C^0(\mathbb{R}^n) \cap L^\infty(\mathbb{R}^n)$ then*

$$u(x, t) = \begin{cases} \int_{\mathbb{R}^n} f(y) G(x - y, t) dy = f * G_t & \text{for } t > 0 \\ f(x) & \text{for } t = 0 \end{cases} \quad (1.3)$$

where G is Gaussian function

$$G(x, t) = \frac{\exp(-|x|^2/4t)}{2^n(\pi t)^{n/2}}, \quad x \in \mathbb{R}^n, t > 0, \quad (1.4)$$

is the solution to (1.2). It is unique under the assumption that

$$\forall T > 0 \quad \exists C_T, a_T > 0 : \quad |u(x, t)| \leq C_T \exp(a_T |x|^2) \quad \text{for } (x, t) \in \mathbb{R}^n \times [0, T]$$

or

$$u(x, t) \geq 0 \quad \text{for } (x, t) \in \mathbb{R}^n \times [0, T]$$

The second condition for uniqueness is particularly interesting, because of its physical interpretation (temperature is never below absolute zero).

It is visible from (1.3), that solving the heat equation means convolving the initial temperature distribution with the Gauss function with the standard deviation parameter $\sigma = \sqrt{2t}$. Such operation is well known as an efficient down-pass spectral filtration and it is commonly used for smoothing pictures by averaging values within a certain neighbourhood. This is where the basic idea of using diffusion equations in image processing comes from.

An important property of the heat equation arises from the properties of convolution:

$$\int_{\mathbb{R}^n} u(x, t) dx = \int_{\mathbb{R}^n} f * G_t dx = \int_{\mathbb{R}^n} f dx \cdot \int_{\mathbb{R}^n} G_t dx = \int_{\mathbb{R}^n} f dx \cdot 1 = \int_{\mathbb{R}^n} u(x, 0) dx \quad (1.5)$$

The interpretation of this result would be the energy conservation law for the transport of heat, or, in image processing, the conservation of the average intensity of the given image.

1.2. Introducing the Perona-Malik equation

Using Gaussian diffusion for image filtration has one great drawback: its homogeneity, which leads to unwanted diffusion of image features like edges. The idea behind the Perona-Malik equation is to modify the heat equation by adding the diffusivity coefficient depending on space activity in a given part of a picture, measured by the norm of the local picture gradient. For small gradient norm (homogeneous regions) large values of the diffusivity are expected, to perform stronger smoothing. In regions with big gradient norm (inhomogeneity) smaller diffusivity is expected, to slow down the diffusion process and protect delicate image features. Hence, the precise Perona-Malik problem formulation with a Neumann boundary condition is

$$\begin{cases} u_t = \nabla(c(|\nabla u|^2)\nabla u) & \text{in } \Omega \times (0, +\infty) \\ \frac{\partial u}{\partial n} = 0 & \text{in } \partial\Omega \times (0, +\infty) \\ u(x, 0) = u_0(x) & \text{in } \Omega \end{cases} \quad (1.6)$$

Here Ω denotes picture domain. In general Ω is a bounded subset of \mathbb{R}^n with boundary of class C^1 . When looking for C^1 solutions, we assume that the derivatives of u_0 vanish at the boundary of Ω . We usually restrict to diffusivity functions $c(s^2)$ monotonically decreasing from 1 to 0 while s^2 changes from 0 to $+\infty$, infinitely smooth and such that function $\Phi(s) = sc(s^2)$ has one maximum in \mathbb{R}_+ (although some authors investigated also different types of diffusivities). The typical choices are

$$c(s^2) = \frac{1}{1 + \frac{s^2}{\lambda^2}} \quad (1.7)$$

and

$$c(s^2) = \exp\left(-\frac{s^2}{2\lambda^2}\right) \quad (1.8)$$

with the parameter $\lambda > 0$.

The flux function is defined as

$$\Phi(s) = s \cdot c(s^2) \quad (1.9)$$

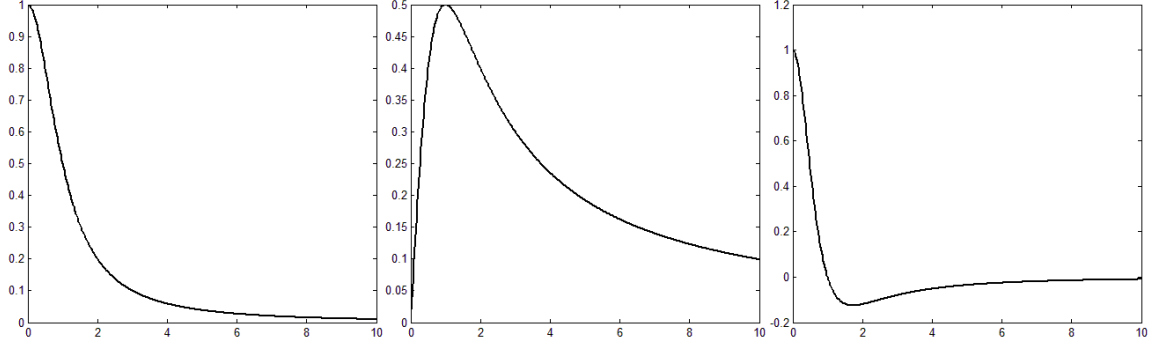


Figure 1.1: Diffusivity function (1.7), the corresponding flux function and its derivative

For diffusivity (1.7) $\Phi'(s) < 0$ for $|s| > \lambda$ and $\Phi'(s) > 0$ for $|s| < \lambda$ (it is an easy calculation to give similar rule for 1.8).

In the one-dimensional case (1.6) simplifies to

$$u_t = \left(c(u_x^2) + 2u_x^2 c'(u_x^2) \right) u_{xx} = \Phi'(u_x) u_{xx} \quad (1.10)$$

Now it is visible, that for large values of u_x the diffusion coefficient $\Phi'(u_x)$ becomes negative, which leads to backward diffusion.

In the two-dimensional case (1.6) can be rewritten as (see [AGLM]):

$$u_t = \Phi'(|\nabla u|) u_{\eta\eta} + c(|\nabla u|^2) u_{\xi\xi} \quad (1.11)$$

where the coordinates η and ξ denote the directions parallel and perpendicular to ∇u respectively. Therefore diffusion is of forward type along the contour lines of the function u – *isophotes* and forward-backward type in the perpendicular direction (gradient direction). We expect edges to be the regions of a large gradient values and therefore of backward diffusion in the gradient direction, which leads to sharpening instead of blurring. This explains how the Perona-Malik filter is not only able to prevent edges from being smoothed, but also enhance them.

We see now that the Perona-Malik model is not anisotropic in the sense that diffusivity is a scalar function, not a tensor field. Nevertheless, since (1.11) shows that diffusivity coefficient varies in perpendicular directions, it is often referred to as an anisotropic diffusion equation.

1.3. Investigation of well-posedness

Definition 1.1 *A well-posed problem is a problem, for which there exists a unique solution, continuously depending on the data (in some reasonable topology). A problem, which is not well-posed is ill-posed.*

Defintion 1.2 (Weak formulation in the one-dimensional case) *A locally integrable function $u(x, t)$ is said to be a weak solution of the Perona-Malik equation if $\int_{\Omega} (u^2 + u_x^2) dx$ is uniformly bounded for bounded t and if for any test function $\phi \in C_0^1(\mathbb{R}_+ \times \mathbb{R})$*

$$\iint [\phi_t u - \phi_x c(u_x^2) u_x] dx dt = 0 \quad (1.12)$$

Backward diffusion is well known to be an ill-posed process. Therefore possible presence of backward diffusion in Perona-Malik model suggests that the equation might be ill-posed. Pessimistic results came from Höllig [Hol], who gave an example of a different forward-backward diffusion process which can have infinitely many solutions. Since then many specific results were found. Below I survey some of the most important ones.

Kichenassamy in [Kit], 1997, showed that if any weak solution of the one-dimensional problem exists, the initial data must be infinitely differentiable in the regions of backward diffusion ($|u_x| > \lambda$). This shows, that there might be no weak solution at all. Of course affine stationary functions ($u(x, t) = ax + b$) are solutions, but they are unstable meaning that there might be no solution if the initial condition is changed arbitrarily little. He also introduced the idea of looking for a generalized solutions of the Perona-Malik equation, showed that the Heavside step function is a generalized stationary solution and constructed a family of piecewise linear solutions.

Definition 1.3 *A function $u(x, t)$ is called a generalized solution (or ultraweak solution) of an equation $P(u) = 0$ in an open set D if there is a sequence u_n of Lipschitz continuous functions such that $P(u_n) \rightarrow 0$ in the sense of distributions in D , and $u_n \rightarrow u$ in $L_{loc}^1(D)$. A generalized solution of an evolution equation is called admissible if its essential variation with respect to the space variables is nonincreasing in time.*

Theorem 1.2 ([Kit] Properties of solutions) *There is no weak solution of problem (1.6) in one dimension, that has bounded derivative which stays greater than λ in a rectangle $(A, B) \times (0, T)$ for any $A, B \in \Omega$, $A < B$, $T > 0$.*

Kawohl and Kutev in [KK], 1998, proved that there are no global C^1 weak solutions of the one-dimensional problem with initial data involving backward diffusion (other than affine functions):

Theorem 1.3 ([KK] Nonexistence of solutions in the one-dimensional case) *Given u_0 such that $u'_0(x) = 0$ in $\partial\Omega$ and $u'_0(x) > \lambda$ (the diffusivity function parameter) in just one compact subinterval of Ω i.e. $u'_0(x) > \lambda$ in $Q = (x_0, y_0) \subset \subset \Omega$ and $|u'_0(x)| < \lambda$ in Ω/\overline{Q} , u_0 analytic near x_0 (or strictly convex), the one-dimensional problem (1.6) has no global weak solution in $C^1(\Omega)$. The result can be extended to finitely many subintervals and $u'_0(x) < -\lambda$.*

They also established a useful maximum and comparison principles for the solutions and proved the uniqueness of local solutions (if they exist). Moreover, they showed that for initial data without backward diffusion there are classical global solutions in C^2 .

Theorem 1.4 ([KK] Maximum principle) *Suppose that u is a Lipschitz continuous (weak) solution of (1.6). Then for every $p \in [2, \infty]$ the following inequality holds:*

$$\|u(x, t)\|_{L^p(\Omega)} \leq \|u_0(x)\|_{L^p(\Omega)} \quad (1.13)$$

Theorem 1.5 ([KK] Comparison principle, one-dimensional case) *Suppose that u and v are local weak solutions of (1.6) with initial data u_0 and v_0 . Then for $t \in (0, T)$: $u_0(x) \leq v_0(x) \Rightarrow u(x, t) \leq v(x, t)$ if any of the following conditions holds:*

a)

$$\{x \in \Omega : |u'_0(x)| > \lambda\} \cap \{x \in \Omega : |v'_0(x)| > \lambda\} = \emptyset$$

b)

$$\begin{aligned} &\exists w_0(x) \in C^{2,\alpha} \text{ such that} \\ &\forall x \in \Omega \ u_0(x) \leq w_0(x) \leq v_0(x), \ |w_0(x)| < \lambda, \ w'_0(x) = 0 \text{ in } \partial\Omega \end{aligned}$$

Theorem 1.6 ([KK] Uniqueness of local weak solutions, one-dimensional case) *Suppose that u and v are local weak solutions of (1.6) in $Q_T = \Omega \times (0, T)$ with identical analytic initial data u_0 , $(u'_0)^2 - \lambda^2$ has only simple zeroes and diffusivity is an analytic function. Then $u(x, t) \equiv v(x, t)$ in Q_T .*

More recently (2006) Rosati and Schiaffino showed [RS] that piecewise linear interpolations of solutions of some discretizations of the Perona-Malik problem converge to ultraweak solutions.

In 2009 Ghisi and Gobbino showed (for diffusivity (1.7)) that the set of initial data for which there exists a local-in-time classical solution is dense in C^1 ([GG]). They also proved an interesting fact that for dimensions higher than one Theorem 1.3 does not hold and the problem admits global solutions of the $C^{2,1}$ class even for the initial conditions indicating backward diffusion ([GG2]).

1.4. Regularizations

The idea behind any regularization of the Perona-Malik equation is to change the basic problem just slightly to obtain a well-posed problem.

In [CLMC] Catte, Lions, Morel and Coll proposed using u spatially convolved with the Gaussian function with standard deviation σ in the diffusion coefficient. This leads to the following problem:

$$\begin{cases} u_t = \nabla(c(|\nabla G_\sigma * u|^2)\nabla u) = \nabla(c(|\nabla u_\sigma|^2)\nabla u) & \text{in } \Omega \times (0, T] \\ \frac{\partial u}{\partial n} = 0 & \text{in } \partial\Omega \times (0, T] \\ u(x, 0) = u_0(x) & \text{in } \Omega \end{cases} \quad (1.14)$$

This problem turns into the Perona-Malik problem for $\sigma = 0$, if we assume $G_0(x) = \delta(x)$ (Dirac's delta distribution). For $\sigma > 0$ there is a unique regular solution of (1.14) (a proof can be found in [CLMC]).

The intuition is that a solution exists because of the smoothing properties of convolution. Although this model became quite common for technical applications, using the convolution with the Gaussian function is questionable, as Perona-Malik process was first proposed to

replace Gaussian filter.

Different approaches involve regularization in time by averaging gradient value within some time interval, introducing a relaxation time into diffusivity [Bar], or mixing spatial and time regularization.

Interesting anisotropic regularization was given by Weickert [Weick], who proposed using a diffusion tensor C instead of a scalar function. C should have two eigenvectors

$$v_1 \parallel \nabla u_\sigma, \quad v_2 \perp \nabla u_\sigma$$

with the eigenvalues

$$\lambda_1(\nabla u_\sigma) = c(|\nabla u_\sigma|^2), \quad \lambda_2(\nabla u_\sigma) = 1$$

The eigenvalue acts as a diffusivity in the respective eigenvector direction. Again, we obtain the Perona-Malik filter as $\sigma \rightarrow 0$ and $C\nabla u \rightarrow c(|\nabla u|^2)\nabla u$.

Chapter 2

Discretizations of the problem

In this chapter I present the results obtained by Weickert [Weick]. These explain how discretization turns the Perona-Malik equation into a well-posed problem.

Definition 2.1 *We call a problem originating from partial differential equation semidiscrete if spatial variables are discrete and time remains continuous. A problem is discrete (or fully-discrete) if the time variable is also discrete.*

2.1. Semidiscrete and discrete models

One possible spatial discretization of (1.6) in two dimensions can be written as (the discussion on discretizing parabolic equations can be found in [MM]), i and j stand for vertical and horizontal pixel indices and with \approx I denote approximating the derivatives by the difference quotients:

$$\left\{ \begin{array}{l} c_{i,j} = c \left(\left(\frac{u_{i+1,j} - u_{i-1,j}}{2\Delta x} \right)^2 + \left(\frac{u_{i,j+1} - u_{i,j-1}}{2\Delta y} \right)^2 \right) \\ c_{i+1/2,j} := \frac{c_{i,j} + c_{i+1,j}}{2} \\ \left[c \frac{\partial u}{\partial x} \right]_{i+1/2,j} \approx c_{i+1/2,j} \left(\frac{u_{i+1,j} - u_{i,j}}{\Delta x} \right) := \varphi_{i+1/2,j} \\ \left[\frac{\partial}{\partial x} \left(c \frac{\partial u}{\partial x} \right) \right]_{i,j} \approx \frac{\varphi_{i+1/2,j} - \varphi_{i-1/2,j}}{\Delta x} \\ \frac{du_{i,j}}{dt} = \left[\frac{\partial}{\partial x} \left(c \frac{\partial u}{\partial x} \right) \right]_{i,j} + \left[\frac{\partial}{\partial y} \left(c \frac{\partial u}{\partial y} \right) \right]_{i,j} \approx \frac{\varphi_{i+1/2,j} - \varphi_{i-1/2,j}}{\Delta x} + \frac{\varphi_{i,j+1/2} - \varphi_{i,j-1/2}}{\Delta y} \\ \frac{du_{i,j}}{dt} \approx \frac{1}{2(\Delta y)^2} [(c_{i,j} + c_{i,j+1})(u_{i,j+1} - u_{i,j}) - (c_{i,j-1} + c_{i,j})(u_{i,j} - u_{i,j-1})] + \dots \\ \dots + \frac{1}{2(\Delta x)^2} [(c_{i,j} + c_{i+1,j})(u_{i+1,j} - u_{i,j}) - (c_{i-1,j} + c_{i,j})(u_{i,j} - u_{i-1,j})] \end{array} \right. \quad (2.1)$$

Therefore, for the Neuman boundary condition, with $N(i, j)$ denoting the indices of elements in the neighbourhood of the element $u_{i,j}$ in the respective direction (a set of two

elements or a single element), we have:

$$\frac{du_{i,j}}{dt} = \sum_{(p,q) \in N_x(i,j)} \frac{c_{i,j} + c_{p,q}}{2(\Delta x)^2} (u_{p,q} - u_{i,j}) + \sum_{(p,q) \in N_y(i,j)} \frac{c_{i,j} + c_{p,q}}{2(\Delta y)^2} (u_{p,q} - u_{i,j}) \quad (2.2)$$

For $u_{i,j}$ in interior of the picture this equals:

$$\begin{aligned} \frac{du_{i,j}}{dt} = & \frac{1}{2(\Delta x)^2} [\overbrace{(c_{i,j} + c_{i-1,j})}^{\beta_{i,j}^-} u_{i-1,j} - \overbrace{(c_{i-1,j} + 2c_{i,j} + c_{i+1,j})}^{\alpha_{i,j}} u_{i,j} + \overbrace{(c_{i,j} + c_{i+1,j})}^{\beta_{i,j}^+} u_{i+1,j}] + \dots \\ & \dots + \frac{1}{2(\Delta y)^2} [\overbrace{(c_{i,j} + c_{i,j-1})}^{\delta_{i,j}^-} u_{i,j-1} - \overbrace{(c_{i,j-1} + 2c_{i,j} + c_{i,j+1})}^{\gamma_{i,j}} u_{i,j} + \overbrace{(c_{i,j} + c_{i,j+1})}^{\delta_{i,j}^+} u_{i,j+1}] \end{aligned} \quad (2.3)$$

Now, if we reorder u to create a vector of length $M \cdot N$ (size of the image), we obtain:

$$\frac{d\mathbf{u}}{dt} = (A_x + A_y)\mathbf{u} = A\mathbf{u} \quad (2.4)$$

with A - a sparse matrix of the size $MN \times MN$ with no more than five non-zero elements in each row ($\alpha_{ij} + \gamma_{ij}$, β_{ij}^+ , β_{ij}^- , δ_{ij}^+ , δ_{ij}^-).

This semidiscrete model can be easily discretized in time, to obtain a fully discrete formulation, by putting

$$\begin{cases} \frac{du_{i,j}}{dt}(n\tau) \approx \frac{u_{i,j}^{n+1} - u_{i,j}^n}{\tau} \\ u_{i,j}^{n+1} = u_{i,j}^n + \tau \left(\frac{\varphi_{i+1/2,j}^n - \varphi_{i-1/2,j}^n}{\Delta x} + \frac{\varphi_{i,j+1/2}^n - \varphi_{i,j-1/2}^n}{\Delta y} \right) \end{cases} \quad (2.5)$$

The scheme above is of explicit type, τ represents timestep. In a compact notation it can be rewritten as

$$\mathbf{u}^{n+1} = [I + \tau A(\mathbf{u}^n)] \mathbf{u}^n \quad (2.6)$$

In [WRV] also a so-called semi-implicit scheme is analysed

$$\mathbf{u}^n = [I - \tau A(\mathbf{u}^n)] \mathbf{u}^{n+1} \quad (2.7)$$

however in this case for each timestep we need to solve a system of equations, and the matrix A , although sparse, would have $2,56 \cdot 10^{10}$ elements for a picture of size 400×400 pixels.

I will present here one more scheme, which is not a direct discretization of the Perona-Malik equation, but of a quite similar problem (2.8). It is important, as it was proposed by Perona and Malik in the paper introducing their equation [PM] and may serve as a simple example of an anisotropic filter.

$$u_t = \nabla \left(\begin{bmatrix} c(u_x^2) & 0 \\ 0 & c(u_y^2) \end{bmatrix} \nabla u \right) \quad (2.8)$$

The difference between the problems is that the Perona-Malik equation would have the same expression, $c(|\nabla u|^2)$, on the whole diagonal.

The discretization (explicit scheme) proposed by Perona and Malik is

$$\left\{ \begin{array}{l} c_{i+1/2,j}^n := c \left(\left(\frac{u_{i+1,j}^n - u_{i,j}^n}{\Delta x} \right)^2 \right) \\ c_{i,j+1/2}^n := c \left(\left(\frac{u_{i,j+1}^n - u_{i,j}^n}{\Delta y} \right)^2 \right) \\ \frac{u_{i,j}^{n+1} - u_{i,j}^n}{\tau} = \frac{1}{2(\Delta x)^2} \left[c_{i+1/2,j}^n (u_{i+1,j}^n - u_{i,j}^n) - c_{i-1/2,j}^n (u_{i,j}^n - u_{i-1,j}^n) \right] + \dots \\ \dots + \frac{1}{2(\Delta y)^2} \left[c_{i,j+1/2}^n (u_{i,j+1}^n - u_{i,j}^n) - c_{i,j-1/2}^n (u_{i,j}^n - u_{i,j-1}^n) \right] \end{array} \right. \quad (2.9)$$

2.2. Well-posedness of discrete models

The discretization (2.1) leads to a nonlinear system of ordinary differential equations

$$\left\{ \begin{array}{l} \frac{d\mathbf{u}}{dt} = A(\mathbf{u})\mathbf{u} \\ \mathbf{u}(0) = \mathbf{u}_0 \end{array} \right. \quad (2.10)$$

In [Weick] a general analysis of such semidiscrete problems is given. Here I present only the main results in the form of a theorem.

Theorem 2.1 ([Weick] Properties of semidiscrete problems) *If the matrix A of a semidiscrete problem in the form (2.1) possesses properties P1 - P5, then the described process satisfies the properties I1 - I4.*

P1 $A(\mathbf{u})$ is a continuously differentiable function of \mathbf{u} .

P2 $A(\mathbf{u})$ is a symmetric matrix.

P3 All row sums of $A(\mathbf{u})$ are zero.

P4 All off-diagonal elements of $A(\mathbf{u})$ are nonnegative.

P5 $A(\mathbf{u})$ is irreducible, which means that for any two indexes i, j there exist a sequence of indices (k_0, k_1, \dots, k_r) such that $k_0 = i, k_r = j, a_{k_p, k_{p+1}} \neq 0$ for $p = 0, 1, \dots, r-1$.

Matrix $A(\mathbf{u})$ obtained in (2.4) possesses the above properties, which is an easy observation.

I1 The problem is well-posed, meaning there is unique solution $\mathbf{u}(t) \in C^1([0, T], \mathbb{R}^{MN})$ for every $T > 0$, depending continuously on the initial value and the right-hand side of the

ODE system (stability). This is implied by P1 (with the Picard-Lindelöf theorem).

I2 *Average grey level invariance (this is a usefull property in practical applications)*

$$\forall t > 0 : \quad \frac{1}{MN} \sum_{k=1}^{MN} (\mathbf{u}_0)_k = \frac{1}{MN} \sum_{k=1}^{MN} (\mathbf{u}(t))_k = \mu$$

This is implied by P2 and P3.

I3 *The extremum principle*

$$\forall t > 0, \quad \forall i \in J : \quad \min_{k \in J} (\mathbf{u}_0)_k \leq (\mathbf{u}(t))_i \leq \max_{k \in J} (\mathbf{u}_0)_k, \quad J = \{1, 2, \dots, MN\}$$

This is implied by P3 and P4.

I4 *Convergence to a constant steady state*

$$\forall i \in J : \quad \lim_{t \rightarrow \infty} (\mathbf{u}(t))_i = \mu$$

This is implied by P2-P5, moreover, certain functions (like the variance of data) are decreasing throughtout the process.

Analysis of the fully discrete case is also given in [Weick]. Once again, I present only the most important conclusion.

Theorem 2.2 ([Weick] Properties of discrete problems) *Scheme (2.6) inherits the properties of the semidiscrete scheme (2.1) (I1 - I4) for*

$$\tau < \frac{1}{\frac{2}{(\Delta x)^2} + \frac{2}{(\Delta y)^2}}$$

which is a stability restriction. Scheme (2.7) is unconditionally stable and therefore always possesses the properties I1-I4.

Theorem 2.2 shows that the discretization of the ill-posed Perona-Malik equation may lead to a well-posed problem and therefore explains the Perona-Malik paradox by showing the stability of numerical implementations.

Chapter 3

Numerical results

3.1. Comparison between Perona-Malik and Gaussian filters

Figure 3.1 presents the evolution of one dimensional data under Gaussian (left column) and Perona-Malik (right column) filters - explicit scheme (2.1).

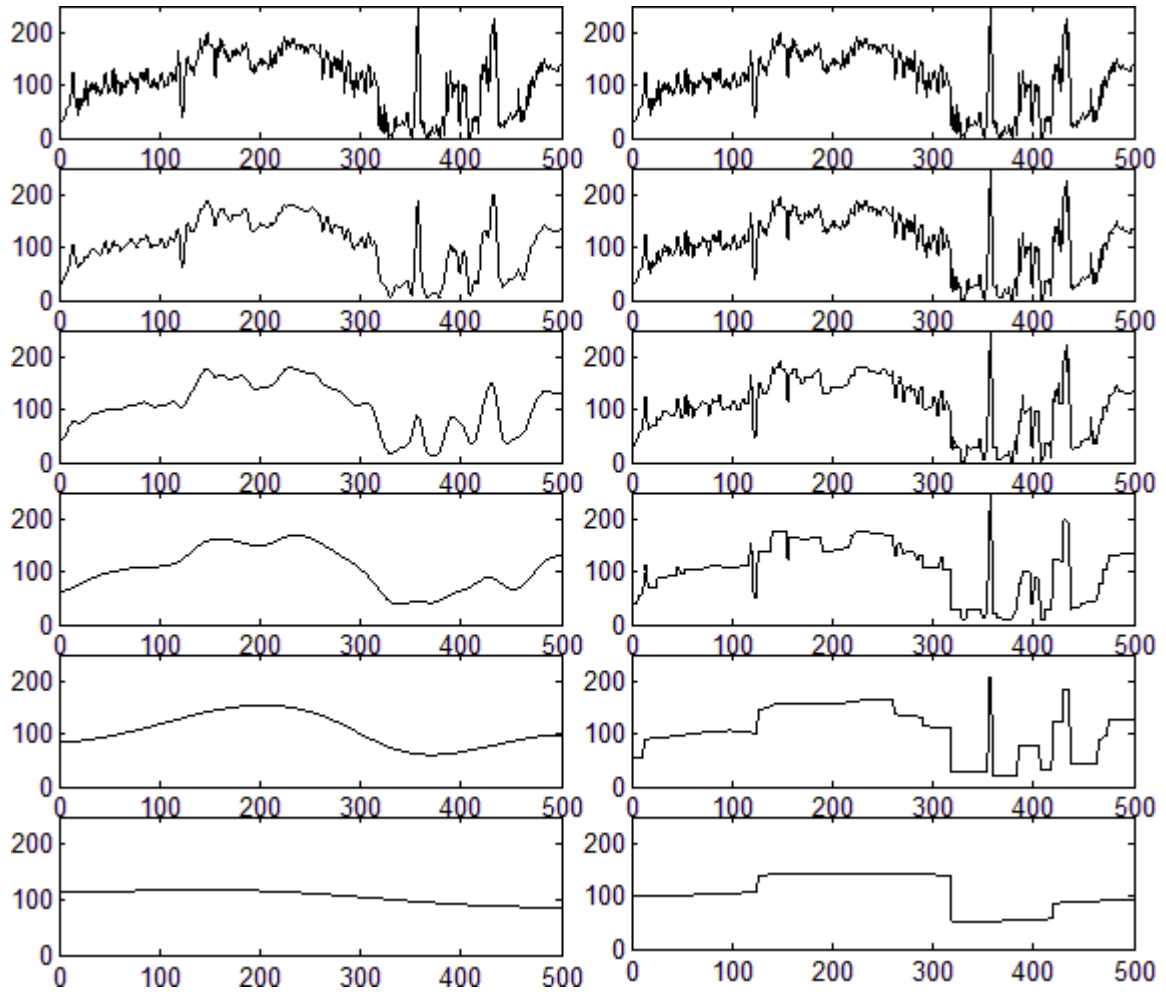


Figure 3.1: Initial data and evolution for 10, 100 , 1000, 10000, 100000 iterations with the timestep 0.1 and $\lambda = 1$.

It is visible on 3.1 how the Perona-Malik filter prevents structures like edges from being smoothed. Another observation is the tendency to develop small regions of constant values of the function.

3.2. Different diffusivity functions

The following figure presents the differences in the evolution of data for different diffusivity functions: model (1.7) (on the left) and (1.8) (on the right) for the explicit scheme (2.1).

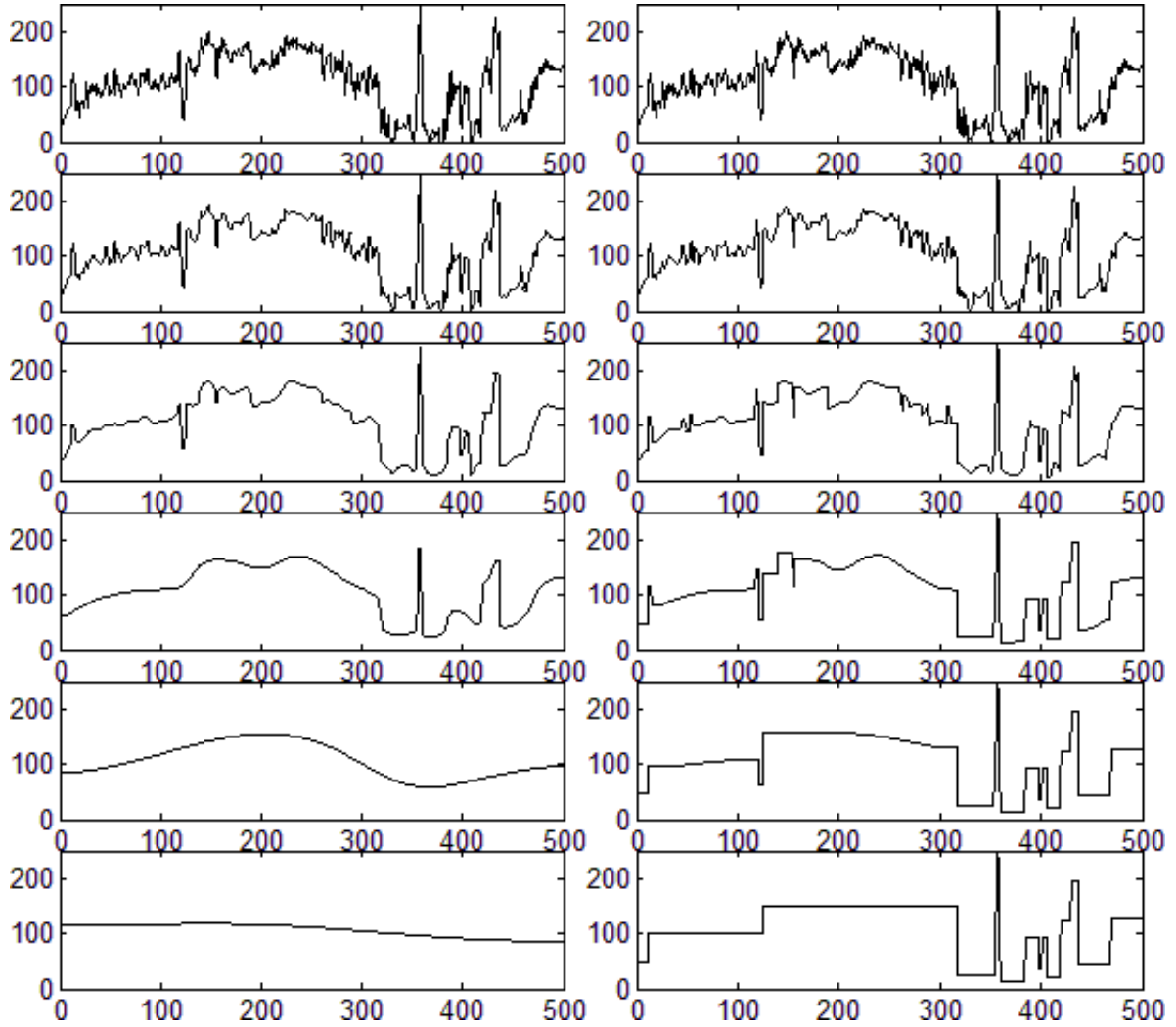


Figure 3.2: Initial data and evolution for 10, 100 , 1000, 10000, 100000 iterations with timestep 0.1 and $\lambda = 5$.

There is a significant difference. The first model acts more like a smoothing filter and flattens the data faster, while the second one preserves the structure for a longer time. This is due to larger negative values of the flux derivative and the enhancing properties of backward diffusion process (fig. 3.3).

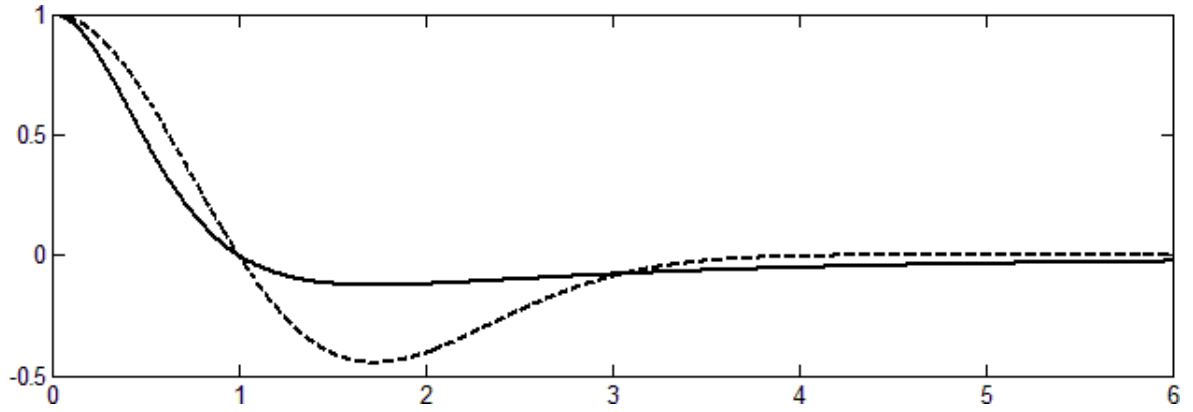


Figure 3.3: Derivatives of the flux function for diffusivities (1.7) (continuous) and (1.8) (dotted) for $\lambda = 1$

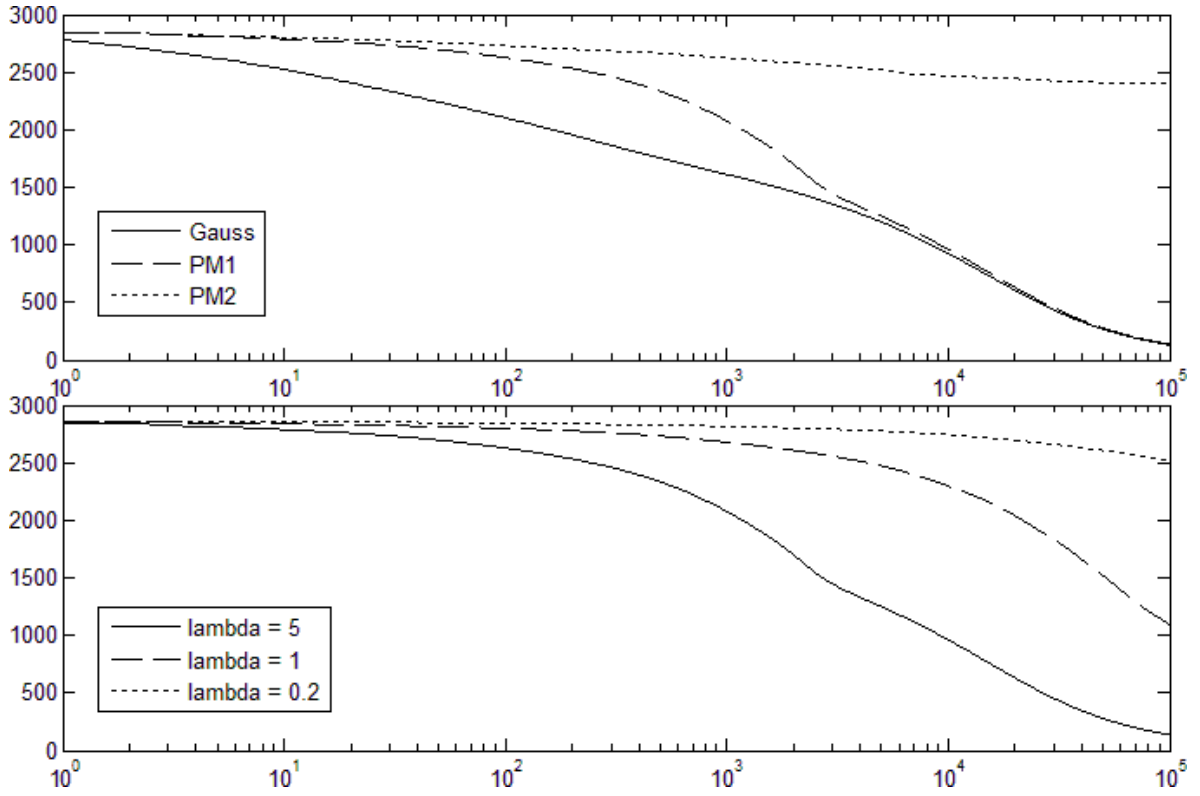


Figure 3.4: Data variance as the function of the number of iterations for processes on fig. 3.2 (PM1 and PM2, respectively) and Gaussian filtration and (lower plot) for diffusivity (1.7) with different λ

3.3. 2D data filtration and evolution

In this section I compare the Perona-Malik filter with the anisotropic process (2.8) and the Perona-Malik filter regularized as in (1.14) with $\sigma = 1$ (pixel). For testing the filtration capability I use a picture with additional Gaussian noise and measure the distance from the

real picture using the metric

$$d(u_0, u) = \sum |u_0(i, j) - u(i, j)| \quad (3.1)$$

Although filtration is not the basic application of the Perona-Malik filter, it is an application for which results are easily measurable. The following figures present the relative error and variance as the function of the number of iterations. Initial data had SNR ratio equal 2 (signal to noise ratio, measured by the mean signal divided by the noise standard deviation). All algorithms were stopped in the first minimum of the error function. The parameter λ was arbitrary chosen to be equal 1, timestep 0.2, diffusivity function (1.7). It took 12690 iterations for the anisotropic algorithm to reach the minimum, 3833 iterations for the standard algorithm and 1557 for the regularized one. Clearly, the regularized algorithm needed much less iterations than the other two and gave slightly better results.

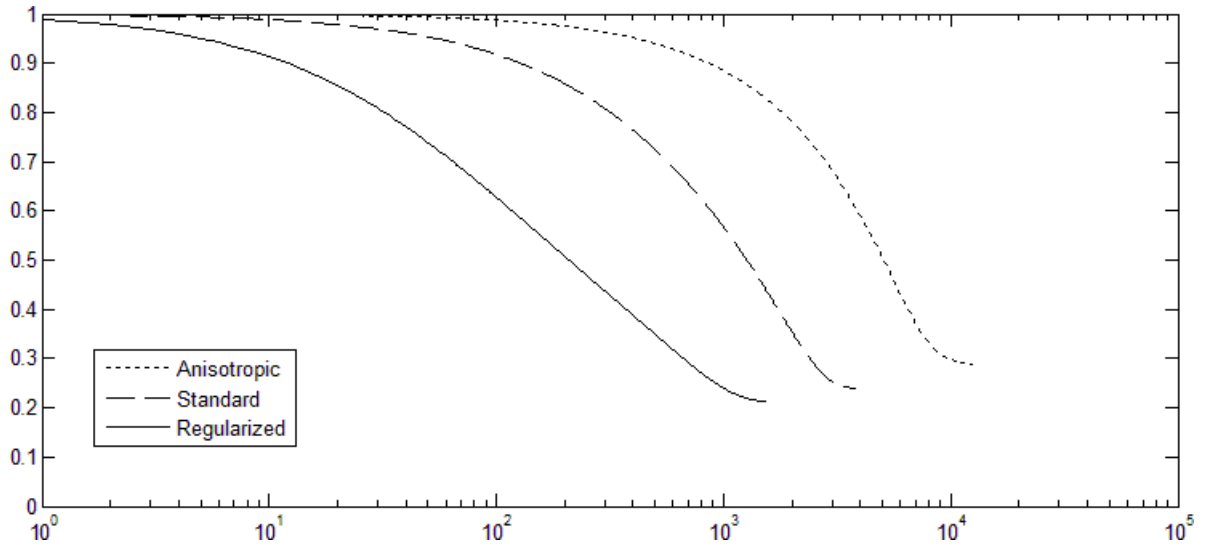


Figure 3.5: Relative error as the function of the number of iterations

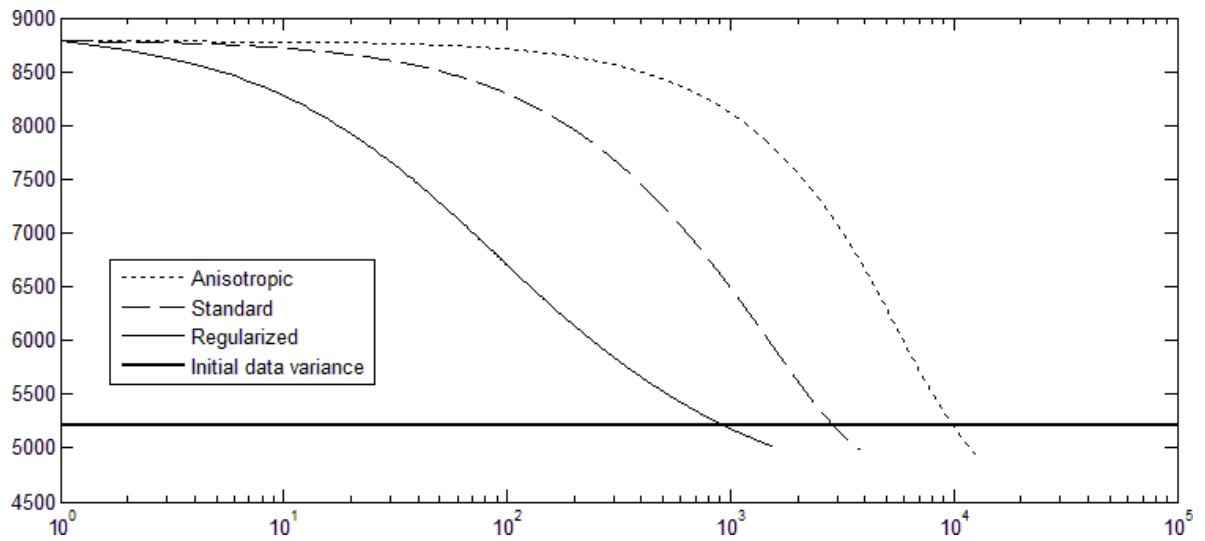


Figure 3.6: Variance as the function of the number of iterations



Figure 3.7: Left to right: picture, picture with noise, picture after 1000 iterations with the standard algorithm, result with the anisotropic algorithm, result with the standard algorithm, result with the regularized algorithm

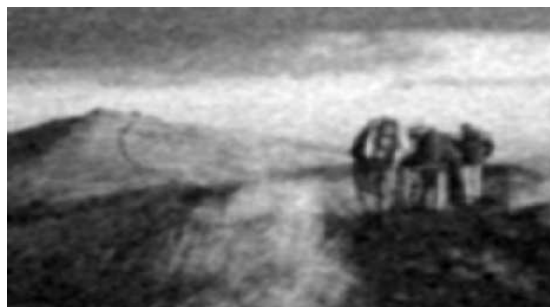


Figure 3.8: For comparison, the effect of Gaussian filtration



Figure 3.9: Evolution of the image "Polonina" under the Perona-Malik filter for 0, 100, 500, 1000, 5000 and 10000 iterations with timestep 0.2 and $\lambda = 1$

It is visible, how the Perona-Malik filter may serve as a tool for image segmentation.

In this chapter as the two-dimensional data I used image "Polonina", bitmap created from a photograph taken by me in Bieszczady mountains in 2009 and as the one-dimensional data I used one horizontal line taken from the image "Polonina".

Appendix A

List of implemented diffusion algorithms

PMexp - Perona-Malik filter with explicit scheme for two-dimensional data

PMexp1D - Perona-Malik filter with explicit scheme for one-dimensional data

PMexpR - regularized Perona-Malik filter with explicit scheme for two-dimensional data

PeronaMalik - Perona-Malik filter in anisotropic version, 2D

PMimp - Perona-Malik filter with semi-implicit scheme, 2D

PMimp1D - Perona-Malik filter with semi-implicit scheme, 1D

GaussDiffusion - simple gaussian diffusion

Bibliography

- [AGLM] Luis Alvarez, Frederic Guichard, Pierre-Louis Lions, Jean-Michel Morel, *Axioms and fundamental equations in image processing*, Archive for Rational Mechanics and Analysis, vol. 123 (1993), 199-257.
- [Bar] G. I. Barenblatt, M. Bertsch, R. Dal Passo, M. Ughi, *A degenerate pseudoparabolic regularization of a nonlinear forward-backward heat equation arising in the theory of heat and mass exchange in stably stratified turbulent shear flow*, SIAM Journal on Mathematical Analysis, vol. 24 (1993), 1414-1439.
- [CLMC] Francine Catté, Pierre-Louis Lions, Jean-Michel Morel, Tomeu Coll, *Image Selective Smoothing and Edge Detection by Nonlinear Diffusion*, SIAM Journal on Numerical Analysis, vol. 29 no. 1. (1992), 182-193.
- [GG] Marina Ghisi, Massimo Gobbino, *A class of local classical solutions for the one-dimensional Perona-Malik equation*, Journal: Transaction of the American Mathematical Society 361 (2009), 6429-6446.
- [GG2] Marina Ghisi, Massimo Gobbino, *An example of global classical solution for the Perona-Malik equation*, arXiv:0907.0772v1 (2009).
- [Hol] Klaus Holig, *Existence of Infinitely Many Solutions for a Forward Backward Heat Equation*, Transactions of the American Mathematical Society, , vol. 278. no. 1. (1983), 299-316.
- [Kit] Satyanad Kitchenassamy, *The Perona-Malik Paradox*, SIAM Journal on Applied Mathematics, vol. 57 no. 5. (1997), 1328-1342.
- [KK] Bernd Kawohl, Nikolai Kutev, *Maximum and comparison principles for one-dimensional anisotropic diffusion*, Mathematische Annalen 311. (1998), 107-123.
- [MM] K. W. Morton, D. F. Mayers, *Numerical Solution of Partial Differential Equations*, Cambridge University Press (1994).
- [PM] Pietro Perona, Jitendra Malik, *Scale-Space and Edge Detection Using Anisotropic Diffusion*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 12. no. 7. (1990), 629-639.
- [RS] Mario Rosatia, Andrea Schiaffinob, *Some remarks about Perona-Malik equation*, Non-linear Analysis 65 (2006), 1-11.
- [Strzel] Paweł Strzelecki, *Krótkie wprowadzenie do równań różniczkowych cząstkowych*, Wydawnictwa Uniwersytetu Warszawskiego, Warsaw, Poland (2006).

- [WB] Joachim Weickert, Brahim Benhamouda, *A Semidiscrete Nonlinear Scale-Space Theory and Its Relation to the Perona-Malik Paradox*, Theoretical Foundations of Computer Vision (1997), 1-10.
- [Weick] Joachim Weickert, *Anisotropic diffusion in image processing*, Teubner-Verlag, Stuttgart, Germany (1998).
- [WRV] Joachim Weickert, Bart M. ter Haar Romeny, Max A. Viergever, *Efficient and Reliable Schemes for Nonlinear Diffusion Filtering*, IEEE Transactions on Image Processing, vol. 7. no. 3. (1998), 398-410.